Research Report

An Initial Investigation of a Modified Procedure for Parallel Analysis

Ou Lydia Liu

Frank Rijmen

Nan Kong

Research & Development

October 2007
RR-07-41

An Initial Investigation of a Modified Procedure for Parallel Analysis

Ou Lydia Liu, Frank Rijmen, and Nan Kong
ETS, Princeton, NJ

October 2007

As part of its educational and social mission and in fulfilling the organization's nonprofit charter and bylaws, ETS has and continues to learn from and also to lead research that furthers educational and measurement research to advance quality and equity in education and assessment for all users of the organization's products and services.

ETS Research Reports provide preliminary and limited dissemination of ETS research. To obtain a PDF or a print copy of a report, please visit:

http://www.ets.org/research/contact.html

Abstract

Parallel analysis has been well documented to be an effective and accurate method for determining the number of factors to retain in exploratory factor analysis. Despite its theoretical and empirical advantages, the popularity of parallel analysis has been thwarted by its limited access in statistical software such as SPSS and SAS, especially in software that analyzes ordinal data. Among the few commonly used procedures, the Hayton, Allen, and Scarpello (2004) procedure requires manually computing the mean of eigenvalues from at least 50 replications. The O'Connor (2000) procedure overcomes that limitation, yet it has difficulties dealing with random missing data. To address these technical issues of parallel analysis for ordinal variables, we adapted and modified the O'Connor procedure to provide an alternative that best approximates the ordinal data by factoring in the frequency distributions of the variables (e.g., the number of response categories and the frequency of each response category per variable). Our procedure has a slightly different theoretical rationale from O'Connor's as well as a practical advantage in dealing with missing data.

Key words: Categorical data, factor analysis, factor retention, parallel analysis, SAS program

Acknowledgments

We would like to thank Dan Eignor, Patrick Kyllonen, Richard Roberts, Yasuyo Sawaki, and Larry Stricker at ETS for their helpful comments, suggestions, and edits on earlier drafts of this report. The authors also gratefully thank Kim Fryer and Jenifer Minsky for their editorial assistance.

Parallel Analysis: An Overview

Factor analysis can be generally characterized as a set of multivariate statistical methods for data reduction and for establishing a more parsimonious relationship among measured variables (Fabrigar, Wegener, MacCallum, & Strahan, 1999). The American Educational Research Association, the American Psychological Association, and the National Council on Measurement in Education (1999) defined it as "any of several statistical methods of describing the interrelationships of a set of variables by statistically deriving new variables, called factors, that are fewer in number than the original set of variables" (p. 175). Factor analysis is also an important tool that is used to investigate the construct validity of many psychological measures. A critical component in the process of construct validation is to determine the number of factors on the basis of exploratory factor analysis (EFA). Compared to confirmatory factor analysis, EFA is particularly useful when there is little theoretical basis for specifying the number of factors or the relationship between the variables and latent factors (Hurley et al., 1997). The selection of the appropriate number of factors is one of the most important decisions in EFA, since *underfactoring* (selecting too few factors) or *overfactoring* (selecting too many factors) can have a deleterious impact on subsequent confirmatory analysis and item estimates (Brown, 2006). Ideally, the number of factors should represent qualitatively distinct constructs that conform to the underlying guiding theory.

Parallel analysis is one of the methods to help determine the number of factors in EFA (Carraher & Buckley, 1995; Horn, 1965). The underlying rational for parallel analysis is that the eigenvalues of the salient factors from real data with a valid latent factor structure should be larger than the eigenvalues of the corresponding factors generated from random data (Ford, MacCallum, & Tait, 1986; Lautenschlager, 1989). Parallel analysis requires the construction of a number of correlation matrices of random variables based on the same sample size and number of variables in the real data set. The mean eigenvalues from the random correlation matrices are compared to the eigenvalues from the real data correlation matrix. For example, the average of the first eigenvalues from the random data is compared to the first eigenvalue from the real data, and the average of the second eigenvalues from the random data is compared to the second eigenvalue from the real data. Namely, the eigenvalues observed from the real data should be larger than the corresponding average eigenvalues from the random data. Otherwise, the observed eigenvalues are considered a consequence of sampling error (Glorfeld, 1995; Horn,

1965). Accordingly, a substantial and meaningful factor should account for more variance than expected by chance alone.

To perform parallel analysis, a number of *k* random data sets should be generated and the average of the eigenvalues from these *k* replications should be compared with the eigenvalues from the real data. There is no set rule regarding the number of random data sets that should be generated for parallel analysis. Horn (1965) suggested that the number should be reasonably large; other researchers suggested from 500–1,000 repetitions (Hayton et al., 2004). However, no significant difference has been found between using one random data set and 100 random data sets (Crawford & Koopman, 1979). Although there is no documented evidence, it appears that 50 data sets are reasonably large enough to perform parallel analysis (Hayton et al.). In terms of evaluating the results, a modification has been proposed by Glorfeld (1995) to Horn's approach using the mean eigenvalues. The modified procedure for parallel analysis replaces the usual average of the eigenvalues by the upper 95^{th} percentile (or any other reasonable percentile) in determining the number of factors. This reduces the tendency to overextract or extract poorly supported factors.

Besides parallel analysis, other factor retention methods are available to determine the number of factors, including the Kaiser-Guttman rule (Guttman, 1954; Kaiser, 1960, 1970), the scree test (Cattell, 1966), the minimum average partial (MAP) method (Velicer, 1976), and the maximum likelihood (ML) estimation. The Kaiser-Guttman rule and the scree test are probably the two most widely used methods based on eigenvalues. The Kaiser-Guttman rule is very straightforward. It specifies that the number of factors to retain equals the number of eigenvalues larger than one. The logic that underlies this rule is that when an eigenvalue is less than one, the variance explained by the factor is smaller than that explained by a single indicator (Brown, 2006). The Kaiser-Guttman rule is the default option in many popular software programs such as SPSS, due to its simplicity. However, the population-based characteristics of this method make it vulnerable to overestimating or underestimating the number of factors to retain (Horn, 1965).

In a population correlation matrix, the eigenvalues for uncorrelated variables are one. When it comes to looking at a finite sample, sampling error and least squares bias cause the eigenvalues to vary from the these values (Hayton et al., 2004). Lance, Butts, and Michels (2006) raised the point that as a widely applied method, the Kaiser-Guttman rule has been documented repeatedly as retaining too many—sometimes far too many—factors, thus introducing

difficulties in determining a reasonable number of factors. In addition, the Kaiser-Guttman rule is intended to provide a lower bound for the rank of the correlation matrix. Therefore, it should be used to specify the upper bound for the number of factors to retain. However, it has been often used to determine the exact number of factors (Gorsuch, 1983).

The scree test is another popular approach in helping to determine the number of factors in EFA. Its central feature is a graph, with a horizontal axis representing the factors and a vertical axis representing the eigenvalues. The graph's function is to help locate the last substantial decline in the magnitude of the eigenvalues. This provides an indication that a major limitation of this approach lies in its subjectivity. It is less of a problem when the sample size is large and the data contain well-defined factors (Gorsuch, 1983). However, when the sample is small, it is less likely that there are apparent or clear shifts in the slope. In this case, the scree test method is prone to subjectivity and possibly ambiguity.

The MAP method advanced by Velicer (1976) is based on partial correlations. This method calculates the average of squared partial correlations after each component has been partialled out. When the minimum average squared partial correlation is reached, no further components are extracted. In this case, the residual matrix closely resembles an identify matrix. Noted by Velicer, this method can be applied with any covariance matrix and can be conceptualized as factors that represent more than one variable. Some FORTRAN programming is available to conduct the MAP procedure (Reddon, 1985).

Compared to the methods based on eigenvalues, the ML method of factor extraction is gaining popularity. The issue of determining the number of factors can be conceptualized as choosing the most appropriate model from a series of factor analysis models that differ in their structure (Fabrigar et al., 1999). The goal is to select a model that accounts for substantially more variance than alternative models. An advantage of the ML method is that it offers a wide range of goodness-of-fit indices that can be utilized to determine the number of factors to retain. Some commonly used fit measures include the likelihood ratio statistic (Lawley, 1940), the reliability coefficient for ML factor analysis solutions (Tucker & Lewis, 1973), the root mean-square error of approximation fit index (RMSEA; Browne & Cudeck, 1992), and the expected cross-validation index (ECVI; Browne & Cudeck, 1989). A primary limitation of the ML method is that it requires a strong assumption of multivariate normality. Inaccurate results may occur if this

assumption is severely violated (Curran, West, & Finch, 1996). In addition, this method is heavily influenced by sample size (Hayton et al., 2004).

Parallel analysis has been well documented to be a robust and accurate method for determining the number of factors to retain. Results from various studies have demonstrated that parallel analysis performed better than the Kaiser-Guttman rule (Silverstein, 1987) and many other procedures. For example, it has been found to be more accurate than the ML method (Humphreys & Montanelli, 1975). Additionally, parallel analysis and MAP were found to be the two most accurate methods when compared to the Kaiser-Guttman rule, scree test, and the chi-square test (Bartlett, 1950). More recently, parallel analysis was again proven to be the most accurate method for determining the number of factors to retain, followed by MAP, with the Kaiser-Guttman rule being the least accurate method (Eaton, Velicer, & Fava, 1999; Zwick & Velicer, 1986).

Glorfeld (1995) pointed out that there is little reason not to use parallel analysis, considering the wealth of evidence favoring this procedure. However, too many researchers rely on the Kaiser-Guttman rule, when parallel analysis should be used instead. Hayton et al. (2004) conducted a review of the studies involving EFA in two major journals between 1990 and 1999, the *Academy of Management Journal* and the *Journal of Management*. They found that among the 142 papers using EFA, 47.2% of the studies used some combination of the Kaiser-Guttman rule and the scree test; 25.4% of the studies solely relied on the Kaiser-Guttman rule, and 5.6% of the studies solely relied on the scree test. Surprisingly, none of the studies reported using other methods for determining the number of factors, such as parallel analysis. Fabrigar et al. (1999) pointed out that the lack of widespread use of this procedure includes a paucity of awareness by researchers, insufficient training in graduate school programs, inadequate coverage in textbooks introducing factor analysis knowledge, and simply the inclination to use the methods most frequently used (e.g., Kaiser-Guttman rule, scree test). Another major reason for the limited use of parallel analysis is the restricted access to readily available software programs that can perform parallel analysis. As stated by Brown (2006, p. 29), “A practical drawback of the [parallel analysis] procedure is that it is not available in major statistical software packages such as SAS and SPSS.”

Among the few available examples of the application of parallel analysis, Hayton et al. (2004) provided syntax for conducting the procedure in SPSS. The syntax takes into

consideration the characteristics of ordinal data represented by Likert-type scales, which is a popular format for psychological surveys. Being similar to the real data is a critical characteristic of parallel analysis. The more closely the random data approximate the real data, except for the correlational structure, the more accurate the results are likely to be. Particularly with ordinal data, the factors to be considered when conducting parallel analysis include the number of categories for each variable (e.g., 4-point, Likert-type scale) and the frequency of each response category per variable. The frequencies are important because the variance structure is, in general, not independent of the mean for nonnormally distributed random variables.

As thorough as the Hayton et al. (2004) procedure is, the syntax has a major limitation: It generates eigenvalues for only one random data set. As described above, in order for this method to be valid, at least 50 replications are required. This means that the user would need to run the syntax 50 times and average the results manually, which could be costly with respect to both time and labor.

O'Connor (2000, 2001) also provided syntax for conducting parallel analysis in SPSS, SAS, and MATLAB. Besides the programs that are available for data from normal distributions on this Web site, the "raw" programs listed on the same page are based on a permutation of the raw data, which is suitable for variables from nonnormal distributions. In particular, each column of the data matrix is permuted, so that the response vector of a new case consists of the responses of different old cases. This way, the dependence structure of the original data set is eliminated in the new data sets, but the marginal frequency distributions are preserved. In addition, the syntax is able to produce mean eigenvalues based on a flexible number of replications, thus relaxing the constraints of the procedure introduced by Hayton et al. (2004).

Purpose of This Study

To further refine the approach for parallel analysis, we adapted the SAS syntax provided by O'Connor (2000) and made some modifications to provide an alternative way of conducting parallel analysis, especially for ordinal data. There are some theoretical and practical differences between the O'Connor (2000) approach and our approach.

On a theoretical level, instead of permuting the columns of the original data set, we proceed by estimating the relative frequencies for each variable in the original data set and subsequently simulate the data for this variable in each new data set by drawing from a multinomial distribution for which the probabilities are given by these relative frequencies.

Contrary to O'Connor's (2000) raw programs, the marginal frequency distributions of the variables in the data sets simulated using this procedure will not be exactly the same as those of the original data, due to sampling error. A consequence is that in our procedure, each case is an independent draw, whereas in O'Connor's (2000) raw program, dependencies between cases are introduced by conditioning on the marginal frequencies (i.e., the values for the last case are determined by the values of the other cases). Since the dependencies vanish with increasing sample size, this theoretical distinction should result in negligible differences when the sample size is large enough.

On a practical level, our procedure deals with random missing data in a more straightforward manner than the O'Connor (2000) approach. For our approach, the frequencies for each variable are computed on all available cases. The simulated data will not contain missing values, which is required by the subsequent SAS IML (interactive matrix language) code. In contrast, O'Connor's (2001) permutation-based approach will yield missing values in simulated data sets, thus violating the requirement of the subsequent SAS IML code. Of course, one could get around the missing data issue through, for example, imputing data for the missing observations. The most obvious way to do so would be to use the marginal relative frequencies of the variables, which is similar to our approach. Furthermore, imputing data will no longer assure that the marginal frequencies of the simulated data exactly match those of the original data. In conclusion, the theoretical difference between these two approaches is very small as sample size approaches infinity. However, our approach has a practical advantage in terms of how it deals with random missing data. The SAS syntax for the parallel analysis procedure developed in this study is described in the following section.

The Current Procedure for Parallel Analysis

The procedure we developed, represented by the SAS syntax in the appendix, is comprised of four major components. Each component is described in detail, followed by an example illustrating the output from the syntax at the very end.

Part I: Importing Data and Generating a Frequency Table

1. Import the real data through specifying the file path and file name.
2. Specify the type of data file using the keyword DBMS. An Excel file was used here.

3. Specify the output file using the keyword OUT.
4. Calculate the frequencies of response categories of the real data using PROC FREQ.
5. Generate the frequency table for the real data.

Part II: Generating Random Data From the Multivariate Distribution and Computing Eigenvalues

1. Specify the number of random cases using Ncases.
2. Specify the number of random variables using Nvars.
3. Specify the number of random data sets using Ndatsets.
4. Specify the number of response categories using Nrespcat.
5. Specify the probability of each response category using Prob.
6. Specify the kind of parallel analysis using Kind, where kind = 1 referring to principal components analysis and kind = 2 referring to principal axis or common factor analysis.
7. Generate random variables from the multinomial distribution.
8. Specify the frequency function that corresponds to the frequency of each response category for the real data.
9. Specify the algorithm for conducting principal component analysis (PCA) or specify the algorithm for factor analysis.
10. The default number of categories for the random data is set to four. Change the ELSE IF command accordingly if the number of categories is some other number.

If the eventual goal is to conduct a PCA on the real data, then parallel analyses using the principal components option in the SAS codes should be used. However, when the eventual goal is to conduct a common factor analysis or principal axis factor analysis on the real data, then the decision whether to use PCA or factor analysis for parallel analysis becomes controversial. As noted by O'Connor (2000), experts do not agree on whether principal component eigenvalues or common and principal axis factor eigenvalues should be used to determine the number of factors. Some experts (e.g., Gorsuch, 1983; Humphreys & Montanelli, 1975) have argued that if the

eventual goal is to conduct a common factor analysis or principal axis factor analysis, then communalities should be placed on the diagonal of a correlation matrix, before the eigenvalues are extracted based on the correlation matrix. Others (e.g., Cattell, 1966) have extracted and examined eigenvalues based on the correlation matrix of PCA and have used these eigenvalues to determine the number of factors for the common and principal axis factor analysis. The latter is the procedure used in the scree tests and in SPSS and SAS factor analysis. Here in our SAS codes, we provide both options for the user to make the decision whether to use PCA or factor analysis.

Part III: Output Options

1. Generate frequency tables.
2. Generate the mean and percentile of the eigenvalues for each component for the *n* random data sets.

An Example of Parallel Analysis for the Learning Strategies Scale

The Learning Strategies Scale developed by the Center for New Constructs at ETS was used as an example to illustrate the use of parallel analysis. This scale consists of 52 items, represented by a 4-point Likert scale. This instrument was developed on the basis of an extensive literature review and was designed to measure multiple aspects of student behavioral, cognitive, and metacognitive strategies in learning. Students were asked to select the response (*strongly agree*, *agree*, *disagree*, or *strongly disagree*) that best described how often in the past year each statement had been true for them. A sample question is, "I use headings when I'm taking notes so that I can find information when I look through them again." Respondents were 238 middle school students from three schools in New Jersey, which constituted sixth graders (39%), seventh graders (24%), and eighth graders (37%). To examine the construct validity of the Learning Strategies Scale, EFA was conducted, using the parallel analysis procedure.

Figure 1 illustrates the results from the EFA using the new procedure for parallel analysis based on 50 replications. Both the mean and the 95 percentiles of the eigenvalues from the simulated random data were included. As shown in Figure 1, the mean eigenvalues and the 95 percentiles are very close. Four factors from the real data had larger eigenvalues than the factors generated from the random data.

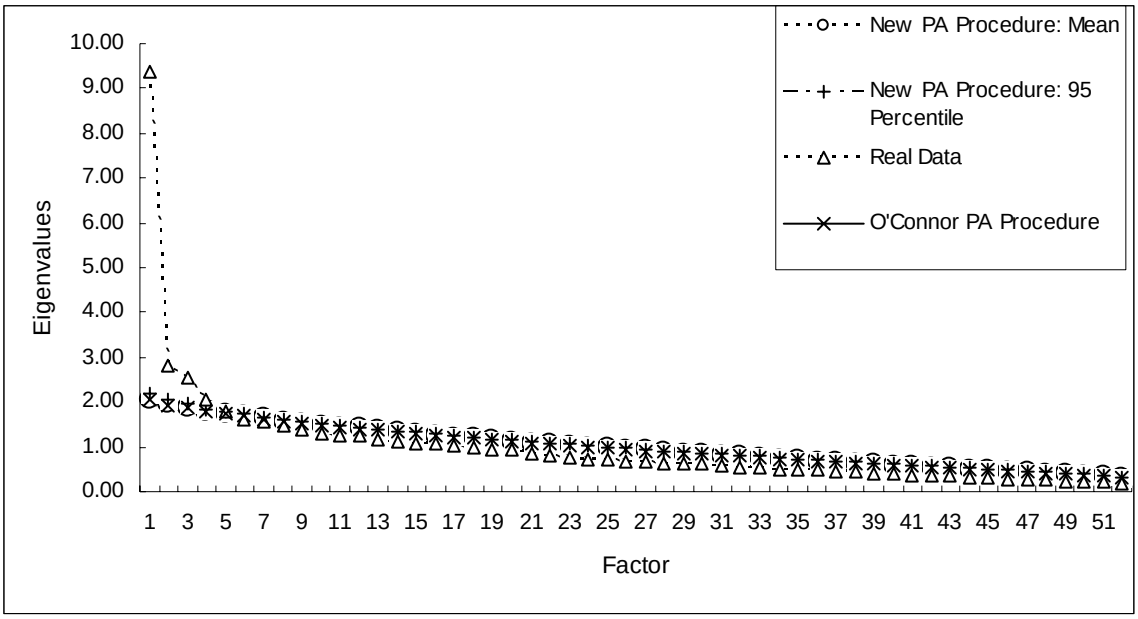

Figure 1. **Results from parallel analysis (PA) for the Learning Strategies Scale.**

The O'Connor (2000) procedure for ordinal data was also tested, and the same number of factors emerged from the results. Table 1 shows the correlations among the eigenvalues from the real-data, mean eigenvalues generated using the O'Connor procedure and the new procedure. In addition, the eigenvalues ordered by size and estimated using the newly developed procedure and the eigenvalues ordered by size and estimated from the O'Connor procedure were in close correspondence, as evidenced by the almost perfect correlation.

Table 1

Correlations Between Real-Data Eigenvalues and Generated Eigenvalues

Eigenvalues	Real-data eigenvalues	O'Connor procedure	New procedure: mean
Real data	1.0000		
O'Connor procedure	0.6831	1.0000	
New procedure: mean	0.6845	0.9998	1.0000
New procedure: 95th percentile	0.6956	0.9996	0.9997

Given the fact that the magnitude of the 4th factor from the real data was slightly greater than the magnitude of the corresponding factor from the random data (Figure 1), a 4-factor

model might be indicated. However, results of a 4-factor EFA model indicated that only two items loaded on the fourth factor, and these two items also loaded saliently on another factor using the larger than .30 loading rule. On the basis of both content analysis and model parsimony, 3 factors were retained for the Learning Strategies Scale, measuring (a) effective strategies, (b) bad habits, and (c) metacognition. It is notable that 17 factors emerged when the Kaiser-Guttman rule was used, which was not very informative in determining how many factors to retain.

Conclusion and Discussion

We hope that this newly developed procedure can provide some practical value in determining the number of factors when conducting EFA. As described above, parallel analysis has been demonstrated to be an effective method for determining the number of factors to retain. The SAS procedure we developed offered a more straightforward solution for categorical data with missing cases. Readers are encouraged to use parallel analysis for factor retention when appropriate, in conjunction with content analysis and other types of evidence. The decision about factor structure should be made on the basis of consistent evidence from multiple sources reflecting both exploratory and confirmatory approaches.

A further refinement of the parallel analysis procedure for ordinal variables would be to compute the eigenvalues on the basis of the polychoric correlation matrix, rather than of the Pearson correlation matrix. However, the computation of a polychoric correlation matrix is computationally intensive, so that conducting a parallel analysis would become cumbersome. Nevertheless, once the number of factors has been selected, further confirmatory factor analyses should be conducted using an estimation procedure that is most appropriate for the data at hand.

References

American Educational Research Association, American Psychological Association, & National Council on Measurement in Education. (1999). *Standards for educational and psychological testing*. Washington, DC: American Educational Research Association.

Bartlett, M. S. (1950). Tests of significance in factor analysis. *British Journal of Psychology, 3,* 77-85.

Brown, T. A. (2006). *Confirmatory factor analysis for applied research.* New York: Guilford Press.

Browne, M. W., & Cudeck, R. (1989). Single sample crossvalidation indices for covariance structures. *Multivariate Behavioral Research, 24,* 445-455.

Browne, M. W., & Cudeck, R. (1992). Alternative ways of assessing model fit. *Sociological Methods and Research, 21,* 230-258.

Carraher, S. M., & Buckley, M. R. (1995). The effect of retention rule on the number of components retained: The case of the Pay Satisfaction Questionnaire. In *Southern Management Association proceedings,* 493-495.

Cattell, R. B. (1966). The scree test for the number of factors. *Multivariate Behavioral Research, 1*, 245-276.

Crawford, C.G., & Koopman, P. (1979). Inter-rater reliability of scree test and mean square ratio test of number of factors. *Perceptual and Motor Skills, 49,* 223-226.

Curran, P. J., West, S. G., & Finch, J. F. (1996). The robustness of test statistics to nonnormality and specification error in confirmatory factor analysis. *Psychological Methods, 1,* 16-29.

Eaton, C. A., Velicer, W. F., & Fava, J. L. (1999). *Determining the number of components: An evaluation of parallel analysis and the minimum average partial correlation procedures.* Unpublished manuscript.

Fabrigar, L. R.,Wegener, D. T., MacCallum, R. C., & Strahan, E. J. (1999). Evaluating the use of exploratory factor analysis in psychological research. *Psychological Methods, 4,* 272-299.

Ford, J. K., MacCallum, R. C., &Tait, M. (1986). The applications of exploratory factor analysis in applied psychology: A critical review and analysis. *Personnel Psychology, 39,* 291-314.

Glorfeld, L.W. (1995). An improvement on Horn's parallel analysis methodology for selecting the correct number of factors to retain. *Educational and Psychological Measurement, 55,* 377-393.

Gorsuch, R. L. (1983). *Factor analysis* (2nd ed.). Hillsdale, NJ: Lawrence Erlbaum.

Guttman, L. (1954). Some necessary conditions for common factor analysis. *Psychometrika, 19,* 149-162.

Hayton, J. C., Allen, D. G., & Scarpello, V. (2004). Factor retention decisions in exploratory factor analysis: A tutorial on parallel analysis. *Organizational Research Methods, 7,* 191-205.

Horn, J. L. (1965). A rationale and test for the number of factors in factor analysis. *Psychometrika, 32,* 179-185.

Humphreys, L. G., & Montanelli, R. G. (1975). An investigation of the parallel analysis criterion for determining the number of common factors. *Multivariate Behavioral Research, 10,* 193-206.

Hurley, A. E., Scandura, T. A., Schriesheim, C. A., Brannick, M.T., Seers, A., & Vandenberg, R. J. (1997). Exploratory and confirmatory factor analysis: Guidelines, issues, and alternatives. *Journal of Organizational Behavior, 18,* 667-683.

Kaiser, H. F. (1960). The application of electronic computers to factor analysis. *Educational and Psychological Measurement, 20,* 141-151.

Kaiser, H. F. (1970). A second-generation little jiffy. *Psychometrika, 35,* 401-415.

Lance, C. E., Butts, M. M., & Michels, L. C. (2006). The sources of four commonly reported cutoff criteria: What did they really say? *Organizational Research Methods, 9*(2), 202-220.

Lautenschlager, G. J. (1989). A comparison of alternatives to conducting Monte Carlo analyses for determining parallel analysis criteria. *Multivariate Behavioral Research, 24,* 365-395.

Lawley, D. N. (1940). The estimation of factor loadings by the method of maximum likelihood. *Proceedings of the Royal Society of Edinborough, 60,* 64-82.

O'Connor, B. P. (2000). SPSS and SAS programs for determining the number of components using parallel analysis and Velicer's MAP test. *Behavior Research Methods, Instrumentation, and Computers, 32,* 396-402.

O'Connor, B. P. (2001). Extension: SAS, SPSS, and MATLAB programs for extension analysis. *Applied Psychological Measurement, 27,* 113-120.

Reddon, J. R. (1985). MAPF and MAPS: Subroutines for the number of principal components. *Applied Psychological Measurement, 9,* 97.

Silverstein, A. B. (1987). Note on the parallel analysis criterion for determining the number of common factors or principal components. *Psychological Reports, 61,* 351-354.

Tucker, L. R., & Lewis, C. (1973). A reliability coefficient for maximum likelihood factor analysis. *Psychometrika, 38,* 1-10.

Velicer, W. F. (1976). Determining the number of components from the matrix of partial correlations. *Psychometrika, 41,* 321-327.

Zwick, W. R., &Velicer, W. F. (1986). Comparison of five rules for determining the number of components to retain. *Psychological Bulletin, 99,* 432-442.

Appendix

The procedure we developed, represented by the SAS syntax below, is composed of four major components. This procedure is specially tailored for questionnaires using ordinal scales. In this procedure, the variables are generated from a multinomial distribution, the number of categories can be freely specified, and the frequency of each response category is approximated.

```
/*-------Part I: Data input------*/


/*read data, specify correct location*/
PROC IMPORT OUT= WORK.example
            DATAFILE= "C:\E_Disk\LSAB_52.xls"
            DBMS=EXCEL REPLACE;
     SHEET="LSAB_52";
     GETNAMES=YES;
     MIXED=NO;
     SCANTEXT=YES;
     USEDATE=YES;
     SCANTIME=YES;
RUN;

/*compute frequencies of response categories and write to SAS dataset*/
ods trace on;
ods output OneWayFreqs=freqs ;
proc freq data=example;
tables V1-V52 ;/*specify variable names here*/
run;


data freqs;
set freqs (keep=Frequency);
run;


        /*-------Part II: Generating variables from a multinomial distribution and Computing
                                    Eigenvalues------*/


options nocenter nodate nonumber linesize=90;   title;
proc iml;

start RANDMULTINOMIAL( N, NumTrials, Prob );
            mP = rowvec(Prob);
```

```
        d = ncol(mP);

        /* check parameters */
        if N<1 then do;
        print "The requested number of observations should be at least 1:" N; stop;
        end;

if NumTrials <1 then do;
        print "The number of trials should be at least 1:" NumTrials; stop;
        end;

        if abs(1 - sum(Prob))>1e-8 then do;
    print "The probabilities must sum to 1:" (sum(Prob))[label="Sum"]; stop;
 end;

        if ncol(loc(Prob>0)) < d then do;
        print "Each probability should be positive:" Prob; stop;
        end;

        b = mP;
        order = d + 1 - rank(mP);
        mP[order] = b; /*For efficiency, sort probabilities in a descending order*/
        X = j(n,d,0);
        z = 0;

        do i = 1 to N;
                if d = 1 then do;
                        X[i] = NumTrials;
                end;
                else do;
                   m = NumTrials;
                        q = 1;
                        call randgen(z,'BINOM',m,mP[1]);
                        X[i,1] = z;
                   do j = 2 to d-1 by 1 while ( m > 0 );  /* m should be positive */
                        m = m - X[i,j-1];
                                q = q - mP[j-1];
                                newp = mP[j]/q;
                                call randgen(z,'BINOM',m,newp);
                                X[i,j] = z;
                        end;
                        X[i,d] = m - z ; /* to have the sum of x_i's = s*/
                end;
        end;
        outX = X;
        outX[ , 1:d] = X[, order]; /* rearrange the order according to the given probabilities*/
```

```
        return(outX);
finish;
store module=RANDMULTINOMIAL;

reset noname;   seed = 1953125;

/*load freqs of response categories into matrix F*/
use freqs ;
read   all into F;


/* enter your specifications here */
Ncases   = 238;
Nvars    = 52;
Ndatsets = 50;
percent  = 95;
Nrespcat=4;/*all response categories have to be observed*/
/* Specify the desired kind of parellel analysis, where:
   1 = principal components analysis
   2 = principal axis/common factor analysis */

kind = 1 ;

/* set diagonal to a column vector   module */
start setdiag(matname,vector);
do i = 1 to nrow(matname);
do j = 1 to ncol(matname);
if (i = j) then;  matname[i,j] = vector[i,1];
end;end;
finish;

/* row sums module */
start rsum(matname);
rsums =j(nrow(matname),1);
do rows = 1 to nrow(matname);
dumr = matname[rows,];
rsums[rows,1]=sum(dumr);
end;
return(rsums);
finish;
/* principal components analysis  */
if kind = 1  then do;
/* computing random data correlation matrices & eigenvalues */
evals = j(Nvars,Ndatsets,-9999);
nm1 = 1 / (Ncases-1);
do nds = 1 to Ndatsets;
```

```
x = j(Ncases,Nvars);
        do a_var = 1 to Nvars;
                b=(a_var-1)*Nrespcat+1;/* read the freqs of the response catgories of variable
a_var*/
                e=a_var*Nrespcat;

                Prob=F[b:e,1]/sum(F[b:e,1]);/*normalize, freqs into relative freqs */

                                y = RANDMULTINOMIAL( Ncases, 1, Prob );

                do a_case = 1 to Ncases;
                  if y[a_case,1] > 0 then
                          x[a_case,a_var] = 1;
                  else if y[a_case,2] > 0 then
                           x[a_case,a_var] = 2;
                  else if y[a_case,3] > 0 then
                          x[a_case,a_var] = 3;
                  else if y[a_case,4] > 0 then
                          x[a_case,a_var] = 4;
                end;
        end;
vcv = nm1 * (t(x)*x - ((t(x[+,])*x[+,])/Ncases));
d = inv(diag(sqrt(vecdiag(vcv))));
evals[,nds] = eigval(d * vcv * d);
end;
end;


/* principal axis/common factor analysis with SMCs on the diagonal  */
if kind = 2 then do;
/* computing random data correlation matrices & eigenvalues */
evals = j(Nvars,Ndatsets,-9999);
nm1 = 1 / (Ncases-1);
do nds = 1 to Ndatsets;
/*x = normal(j(Ncases,Nvars)) ;*/

x = j(Ncases,Nvars);
        do a_var = 1 to Nvars;
                b=(a_var-1)*Nrespcat+1; /* read the freqs of the response catgories of variable
a_var*/
                e=a_var*Nrespcat;

                Prob=F[b:e,1]/sum(F[b:e,1]); /*normalize, freqs into relative freqs*/

                y = RANDMULTINOMIAL( Ncases, 1, Prob );
```

```
                do a_case = 1 to Ncases;
                  if y[a_case,1] > 0 then
                          x[a_case,a_var] = 1;
                  else if y[a_case,2] > 0 then
                           x[a_case,a_var] = 2;
                  else if y[a_case,3] > 0 then
                          x[a_case,a_var] = 3;
                  else if y[a_case,4] > 0 then
                          x[a_case,a_var] = 4;
                end;
        end;



vcv = nm1 * (t(x)*x - ((t(x[+,])*x[+,])/Ncases));
d = inv(diag(sqrt(vecdiag(vcv))));
r = d * vcv * d;
smc = 1 - (1 / vecdiag(inv(r)) );
run setdiag(r,smc);
evals[,nds] = eigval(r);
end;
end;


/* identifying the eigenvalues corresponding to the desired
percentile */
num = round((percent*Ndatsets)/100);
results = j(nvars,3,-9999);
s = 1:Nvars;
results[,1] = t(s);
do root = 1 to Nvars;
ranks = rank(evals[root,]);
do col = 1 to Ndatsets;
if (ranks[1,col] = num) then do;
results[root,3] = evals[root,col];
col = Ndatsets;
end;
end;
end;
results[,2] = evals[,+] / Ndatsets;


                        /*-------Part III: Output options------*/


print, "Parallel Analysis:";
```

```
if (kind = 1) then;  print, "Principal Components";
if (kind = 2) then do;
print "Principal Axis / Common Factor Analysis";
print "Compare the random data eigenvalues below to the";
print "real-data eigenvalues that are obtained from a";
print "Common Factor Analysis in which the # of factors";
print "extracted equals the # of variables/items, and the";
print "number of iterations is fixed at zero (maxiter=0), as in:";
print "proc factor data=trial  priors=smc maxiter = 0; run; ";
print "Or use the 'rawpar.sas program' to obtain the ";
print "baseline real-data eigenvalues.";
end;
specifs = (ncases // Nvars // Ndatsets // percent);
rlabels = {"Ncases" "Nvars" "Ndatsets" "Percent"};
print/  "Specifications for this Run:", specifs[rowname=rlabels];
clabels={"Root" "Means" "Prcntyle"};
print  "Random Data Eigenvalues", results[colname=clabels format=12.6];

quit;
```